FORECASTING LONG-TERM GOVERNMENT BOND YIELDS: AN APPLICATION OF STATISTICAL AND AI MODELS

MARCO CASTELLANI
CENTRIA – DI
Faculdade Ciências e Tecnologia
Universidade Nova Lisboa
Quinta da Torre,
2829-516 Caparica, Portugal
email: mcas@ftc.unl.pt

EMANUEL AUGUSTO DOS SANTOS[1,2]
email: emanuelagsa@sapo.pt

[1] Affiliation: This paper was mainly prepared in the first half of 2005 when I was member of the Board of IGCP – Portuguese Public Debt Agency
[2] I am grateful to António Afonso for very helpful comments.which benefited the paper. The usual disclaimer applies.

ABSTRACT

This paper evaluates several artificial intelligence and classical algorithms on their ability of forecasting the monthly yield of the US 10-year Treasury bonds from a set of four economic indicators. Due to the complexity of the prediction problem, the task represents a challenging test for the algorithms under evaluation. At the same time, the study is of particular significance for the important and paradigmatic role played by the US market in the world economy. Four data-driven artificial intelligence approaches are considered, namely, a manually built fuzzy logic model, a machine learned fuzzy logic model, a self-organising map model and a multi-layer perceptron model. Their performance is compared with the performance of two classical approaches, namely, a statistical ARIMA model and an econometric error correction model. The algorithms are evaluated on a complete series of end-month US 10-year Treasury bonds yields and economic indicators from 1986:1 to 2004:12. In terms of prediction accuracy and reliability of the modelling procedure, the best results are obtained by the three parametric regression algorithms, namely the econometric, the statistical and the multi-layer perceptron model. Due to the sparseness of the learning data samples, the manual and the automatic fuzzy logic approaches fail to follow with adequate precision the range of variations of the US 10-year Treasury bonds. For similar reasons, the self-organising map model gives an unsatisfactory performance. Analysis of the results indicates that the econometric model has a slight edge over the statistical and the multi-layer perceptron models. This suggests that pure data-driven induction may not fully capture the complicated mechanisms ruling the changes in interest rates. Overall, the prediction accuracy of the best models is only marginally better than the prediction accuracy of a basic one-step lag predictor. This result highlights the difficulty of the modelling task and, in general, the difficulty of building reliable predictors for financial markets.

Keywords: interest rates, forecasting, neural networks, fuzzy logic

NOTATION

AI	Artificial intelligence
ANN	Artificial neural network
BP	Backpropagation
CBOE	Chicago Board Options Exchange
CPI	Consumer Price Index
EA	Evolutionary algorithm
ECM	Error correction model
Fed	Federal Reserve
FL	Fuzzy logic
GDP	Gross domestic product
ISM	Institute for Supply Management)
KB	Knowledge base
Libor	London Inter Bank Offering Rate
MF	Membership function
MLP	Multi-layer perceptron
PMI	Purchasing Managers' Index
RB	Rule base
RMSE	Root mean square error
SOM	Self-organising map

VAR (vector autoregressive)

VIX	Volatility Index

1. INTRODUCTION

Changes in interest rates are important to macroeconomic analysis and economic growth. However, it is in the financial markets that they have a more valuable impact and are most closely monitored. That is so because interest rates are the price for borrowing money and determine the value of financial assets.

Starting from the real world of business, a large amount of information on future and forward contracts on bonds can be collected and used for building a model for the so-called term-structure of interest rates[3]. In a framework of certainty equilibrium, forward rates[4] must coincide with future spot rates[5]. In theory, using a model that maximises the economic agents behaviour under rational expectations, it is possible to get specific formulas, which can be calibrated and empirically tested and used for predicting interest rates[6].

In reality, such an environment does not exist, future interest rates reflect human expectations on many factors not under control. Moreover, given the increasing internationalisation of economies and financial markets, the prediction of interest rates has become more complex, since developments in one country influence other countries as well.

Classical financial modelling theory is based on accurate mathematical identification of the observed system behaviour, modelling and forecasting economic variables using classic econometrics (Greene, 2003) or time series theory (Newbold, 1986; Clements, 1998). Econometrics departs from the specification of a theoretical relationship between a specific economic variable (endogenous) and a set of explanatory variables. In most cases, the postulated functional form can be a linear or non-linear function. The unknown parameters of the model are then estimated using algebraic techniques such as least squares. The estimated model is an eligible tool for making forecasts that can be statistically evaluated.

This structural approach to time series modelling makes use of economic theory to define the structure that is estimated by statistical techniques. Conversely, univariate ARIMA models (Box and Jenkins, 1976) employ pure statistical methods for estimating and forecasting future values of a variable. In this case, current and past values are the only data used in the estimation process.

Unfortunately, the complexity of financial markets and the intrinsic uncertainties regarding their dynamics make the expression of precise analytical relationships often impossible, impractical or

[3] The term-structure of interest rates measures the relationship among the yields of risk-free securities that differ only in their term to maturity.
[4] Forward rates apply to contracts for delivery at some future date.
[5] Spot rates, on the contrary of forward rates, apply to contracts for immediate delivery
[6] The Cox, Ingersoll, Ross (1985) model is an example of an equilibrium asset pricing model for the term structure of interest rates.

just unmanageably complex. Moreover, due to the non-linear, non-parametric nature of economic systems, standard linear econometric modelling has often turned out to be unsatisfactory.

The study of biological nervous systems has shown that highly accurate and robust mappings can be achieved by learning appropriate sets of condition-response pairs. In the field of artificial intelligence (AI) two main approaches have emerged, each modelling cognitive processes at different levels of abstraction. The first method focuses on high-level symbolic associations and expresses complex stimulus-response relationships through sets of *if-then* rules. Fuzzy logic (FL) is a symbolic AI paradigm that extends Aristotle's classical logic to take into account the uncertainty about real world knowledge (Pham and Li, 2005).

The second approach postulates that the computational capabilities of living nervous systems are based on the parallel distributed processing of massively connected networks of simple computing units. Artificial neural networks (ANNs) represent the connectionist AI effort to model the architecture of biological information processing systems (Norgaard et al., 2000).

ANNs and FL systems share common features and complementary limitations. Both paradigms provide a choice of mapping algorithms capable to perform model-free identification of any arbitrarily complex non-linear function (Goonatilake and Khebbal, 1995; White and Sofge, 1992). The approximate nature of their pattern matching and association processes makes them particularly suitable to deal with ill-defined and uncertain problem domains.

These two AI approaches appear as alternatives to modelling and forecasting economic variables using classic theory. This paper compares the performance of AI models and classical econometric and ARIMA models for forecasting the US 10-year Treasury bonds yields. The task is chosen because of its complexity as a modelling problem and because of the role played by the US economy in the world market. A complete series of end-month US 10-year Treasury bonds yields and economic indicators covering 19 years between 1986:1 and 2004:12 are available. The models are fitted using data regarding the first 18 years and evaluated on their capability of forecasting the US 10-year Treasury bonds yields for the remaining 12-months. The root mean square error (RMSE) of the 12-month out of sample forecasts is used to measure the modelling accuracy.

The remainder of the paper is organized as follows. Section 2 presents the problem domain. Section 3 introduces the FL and ANN models. Section 3 describes the econometric and ARIMA models. Section 4 presents the experimental results and compares the performance of the models. Section 5 discusses the results. Section 6 concludes the paper and proposes areas for further investigation.

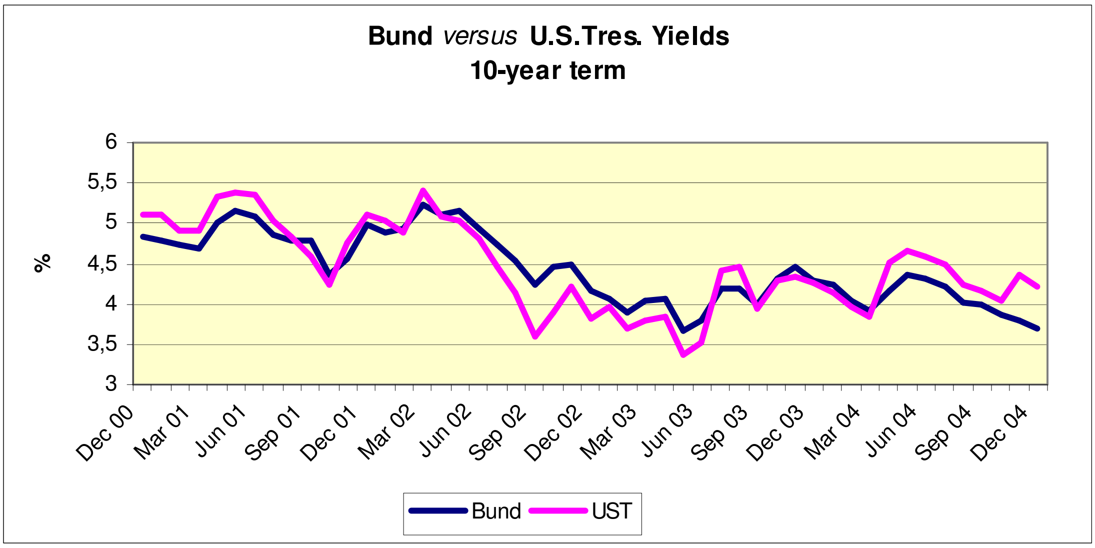


Fig. 1: US treasuries yields versus German bunds yields

2. PROBLEM DOMAIN AND EXPERIMENTAL DATA

The proposed case study concerns the forecasting of the US Treasury bonds yields from the measures of four economic indicators. A complete set of 228 monthly data covering the 19 years between 1986:1 and 2004:12 are available. There are no missing attributes.

The AI and classical approaches presented respectively in Section 2 and Section 3 are evaluated on their accuracy of predicting the correct monthly figure for the US Treasury bonds yields. This figure must be estimated based on the corresponding monthly figure of the four economic indicators. The forecasting task requires the identification of the input-output relationship between the dependent variable (the US bonds yields) and the four independent variables (the indicators). 216 data samples relative to the first 18 years are used to fit the models, and the remaining 12 data samples (2004:1 to 2004:12) are used to evaluate the modelling accuracy.

The choice of forecasting US Treasury bonds yields has its rationale in the fact that the US economy is a paradigmatic market playing an important role in the world economy. In particular, the developments in the US economy have impact on the other two main economic areas – Europe and Japan.

In the case of Europe, it is recognised the existence of a significant correlation between the yields of US treasuries and the yields of German bunds given a stable exchange market. Fig. 1 visualises this correlation during the period 2000:12-2004:12.

Since the German bund is nowadays the benchmark for the bonds issued by the other countries in the euro area, the forecast of US long-term interest rates could help to foresee the future evolution of interest rates on sovereign debt in any other European country. A recent study (Baele, 2004) points out that the government bonds yields in countries belonging to the euro area are sensitive to regional and global shocks but not to idiosyncratic shocks, supporting the assumption of an increasing interrelationship of the financial markets at world level.

2.1. Dependent variable

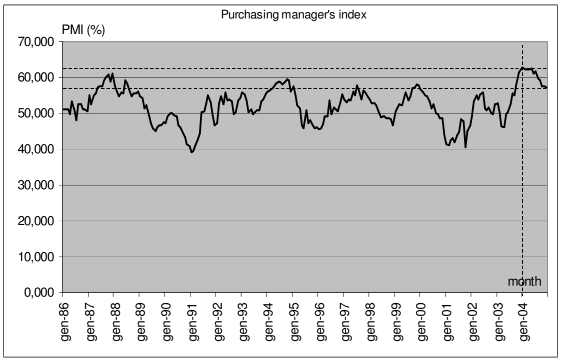
The 10-year U.S. Treasury bonds is one of the fixed maturity securities for which the U.S. Treasury calculates a daily yield. The other maturities are, currently, 1, 3 and 6 months and 1, 2, 3, 5, 7 and 20 years. The 10-year maturity is selected because it is a widespread benchmark used in financial markets. In spite of not being available every day a bond with a constant maturity, it is possible to calculate a theoretic yield of such bond by interpolating the daily yield curve for Treasury nominal securities. The data is the end-period yield for each month disclosed by the Federal Reserve.

2.2. Independent variables

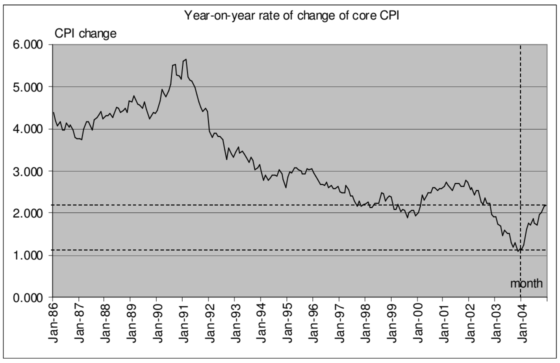
Four economic indicators are chosen as explanatory variables to predict the US Treasury bonds yields, namely, the Purchasing Managers' Index (PMI), the Consumer Price Index (CPI), the London Inter Bank Offering Rate (Libor) and the Volatility Index (VIX).

The economic situation is important to interest rates. When the economy is booming and there is a high demand for funds, the price of borrowing money goes up, leading to increasing interest rates. Conversely, in economic recessions, everything else being equal, there is downward pressure on interest rates. The most important economic indicator for the output of goods and services produced in a country is the gross domestic product (GDP). However, this indicator is published only on a quarterly and annual basis. The PMI published monthly by the ISM (Institute for Supply Management) appears to be a good proxy for the GDP, as it generally shows a high correlation with the overall economy. For example, according to ISM analysis, a PMI in excess of 42.7 percent, over a period of time indicates an expansion of the economy. This month-to-month indicator is a composite index based on the following five indicators for the manufacturing sector of the U.S. economy: new orders, production, employment, supplier deliveries and inventories.

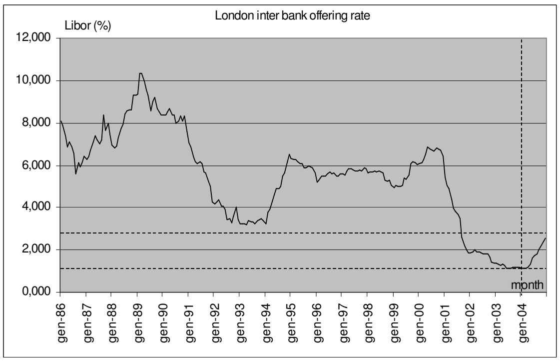
Inflation is important to interest rates as well. Higher-than-expected inflation can cause yields and interest rates to rise, as investors want to preserve the purchasing power of their money. The most important measure of inflation is the average change over time in prices included in the CPI. A more accurate measure of the underlying rate of inflation is obtained when the volatile food and energy prices are excluded from the CPI. The latter measure, sometimes referred as the "core" CPI, is selected for this study as one of the four explanatory variables. The year-on-year rate of change is used in place of the raw core CPI index. The source of the data is the Bureau of Labor Statistics.



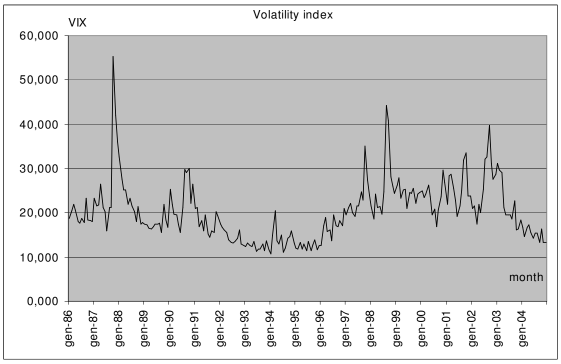
a) PMI index (input variable)



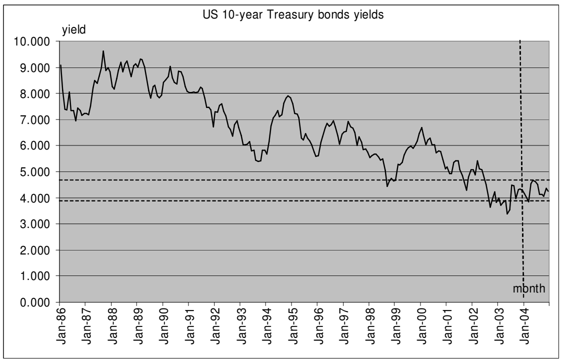
b) CPI index (input variable)



c) LIBOR index (input variable)



d) VIX index (input variable)



e) US 10-year Treasury bonds yields (output variable)

Fig. 2: Input and output variables

Another major factor in interest rate changes is the monetary policy of central banks. For example, the Federal Reserve (Fed) increases or decreases the Fed Funds rate – the key-rate for lending money to the other banks – according to the economic condition. When the economy is growing above its potential and unemployment is low, a central bank will increase rates to curb inflationary pressures. In a recession, a central bank will cut rates to stimulate economic growth and reduce unemployment. In this study the Libor is used instead of the Fed Funds rate for the three-month term. The Libor is an average of the interest rate on dollar-denominated deposits traded between banks in London. The Libor reflects every change in the Fed Funds rate and has the advantage of having a daily market-driven fixing. As the source of data the British Bankers Association is used.

Main statistical measures					
Sample: 1986:01 2004:12 Observations: 228					
	US 10y TB yields	PMI index	CPI-yoy	Libor-3m	VIX index
Mean	6.507105	52.175439	3.153809	5.291756	20.231899
Standard Deviation	1,540538	5.011609	1.075363	2.270260	6.555111
Kurtosis	-0.955555	-0.233122	-0.855340	-0.625088	4.094172
Skewness	0.019810	-0.251113	0.341810	-0.207222	1.455191
Correlation Matrix					
	US 10y TB yields	PMI index	CPI-yoy	Libor-3m	VIX index
US 10y TB yields	1				
PMI index	-0.002433	1			
CPI-yoy	0.86356	-0.309878	1		
Libor-3m	0.842462	-0.183082	0,696916	1	
VIX index	-0.101080	-0.105170	-0.061067	0.070075	1

Table 1: Summary of data.

Another factor that affects the course of bonds yields is the stock exchange condition. When the demand in the capital market shifts from government bonds to equities, bonds prices tend to decrease and bonds yields to increase as these variables move in opposite direction. To capture this relationship an indicator for the stock market volatility is chosen for this study. The VIX compiled by the Chicago Board Options Exchange (CBOE) is chosen. The VIX is calculated using options on the S&P 500 index, the widely recognised benchmark for U.S. equities. The VIX index measures market expectations of near-term volatility and has been considered by many to be the world's premier barometer of investor sentiment. To obtain a long series starting in 1986:1 two indices have to be reconciled: the VOX (1986:1 to 1989:12) and VIX (1990:1 to 2004:12). For the whole period, the most recent indicator VIX (1990:1 to 2004:12) is kept as released by the CBOE and its value for the period 1986:1 to 1989:12 is calculated by using the implicit rates of change in the old series.

Figs. 2a-e show the evolution of the 10-year U.S. Treasury bonds yields together with the evolution of the four explanatory variables over the 19 years period. For each plot, the vertical dashed line marks the division between the 18-years modelling samples and the one-year evaluation samples. The two horizontal lines show the range of variation of the variable over the evaluation period. Table 1 summarises the main statistical measures of the time series.

3. AI MODELLING APPROACHES

There is a large and ever growing literature regarding applications of AI techniques to financial problems. Due to their capability of learning complex non-linear relationships from raw numerical data, ANN systems were often used for prediction of financial time series. Typical applications include the forecasting of interest rates (Cheng et al., 1996; Din, 2003), stock market predictions (Refenes et. al, 1997; Dunis and Jalilov, 2002; Bartlmae et al., 1997), forecasting of currency exchange rates (Walczak, 2001; Chen and Leung, 2005), house pricing and bond rating (Daniels et. al., 1999), etc.. For a broad overview on the use of ANNs in finance, the reader is referred to Trippi and Turban (1996) and McNelis (2005).

ANNs can be divided into *supervised* and *unsupervised*, according to the training procedure implemented (Lippmann, 1987). Supervised ANNs are trained under the control of an omniscient teacher that gives the input and the correct output to be modelled. This is by far the type of ANN most commonly used in financial prediction tasks. Unsupervised ANNs are left free to organise themselves to find the best partition of the input space. In this study, two of the best known examples of of supervised and unsupervised ANN models are evaluated. For a quick introduction to ANN functioning and terminology, the reader is referred to appendix A.

The lack of a standard data induction algorithm makes the implementation of FL system less straightforward. Nonetheless, several studies address the application of FL to financial modelling and decision making (Goonatilake et al., 1995; Mohammadian and Kingham, 1997). In this study, a manually designed FL model and an automatically generated FL system are evaluated. For a quick introduction to FL functioning and terminology, the reader is referred to appendix B.

For all the machine learned AI models, accuracy results are estimated on the average of 10 independent learning trials

3.1. Supervised artificial neural network model

The multi-layer perceptron (MLP) (Lippmann, 1987) is perhaps the best known and most successful type of ANN. It is characterised by a fully connected feedforward architecture composed of three or four layers of processing elements. Fig. 3 shows a typical MLP architecture.

The basic unit of this ANN is the perceptron. The perceptron performs a weighted summation of the input signals and transforms it via a non-linear stepwise transfer function. Fig. 4 shows a perceptron unit.

The input layer fans the incoming signals out to the neurons of the next layer. Since the monthly forecasts for the US Treasury bonds are based on four economic indicators, four input neurons are used.

One or more hidden layers of perceptrons split the input space into several decision regions, each neuron building onto the partition of the previous layer. The more hidden layers there are and the larger they are, the more complex the overall mapping is. However, it can be shown that no more

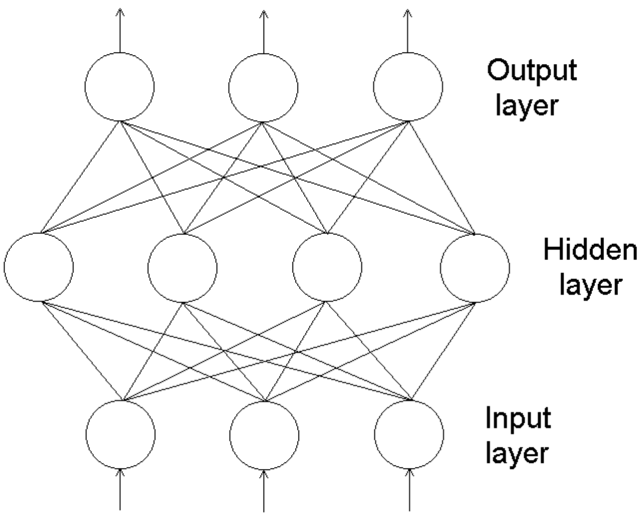


Fig. 3: Multi-layer perceptron

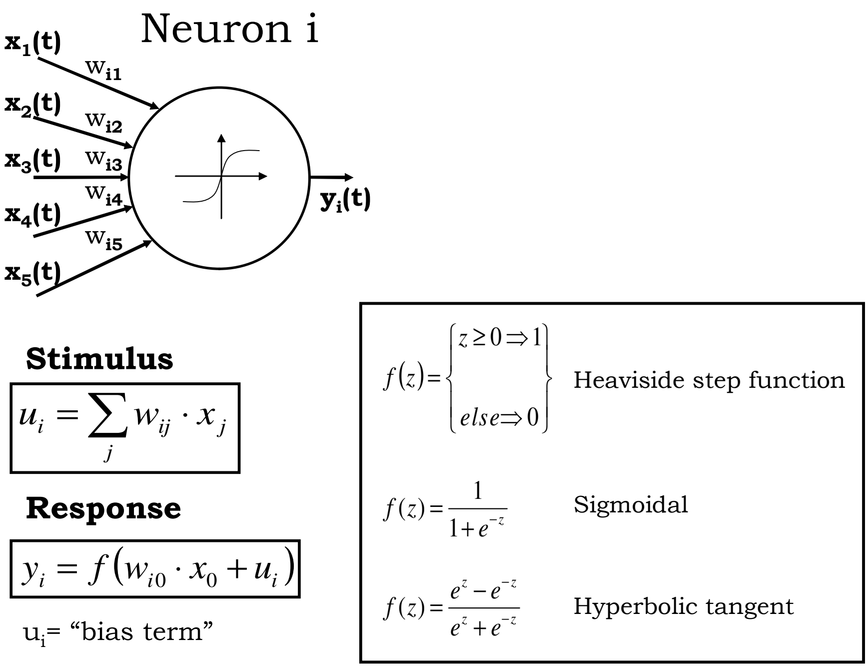


Fig. 4: Perceptron unit

than two hidden layers are required to model any arbitrarily complex relationship (Lippmann, 1987). The optimal configuration is chosen by trial and error, that is, by training different MLP structures and assessing their merit on the learning accuracy. The best prediction results for the US Treasury bonds yields are obtained using one hidden layer of 50 units.

Multi-Layer Perceptron Settings	
Input nodes	4
Output nodes	1
Hidden nodes	50
Activation function of hidden layer nodes	Hyper-tangent
Activation function of output layer nodes	Sigmoidal
Initialisation range for MLP weights	[-0.3, 0.3]
Backpropagation Rule Settings	
Learning coefficient	0.06
Momentum term	0.1
Learning trials	10
Learning iterations	100

Table 2: MLP settings and BP parameters.

The ouput layer collects the signals from the last hidden layer and further processes them to give the final ANN response. Since only one output is required (that is, the monthly forecast for the US treasury bonds yields), this layer is composed of a single perceptron unit.

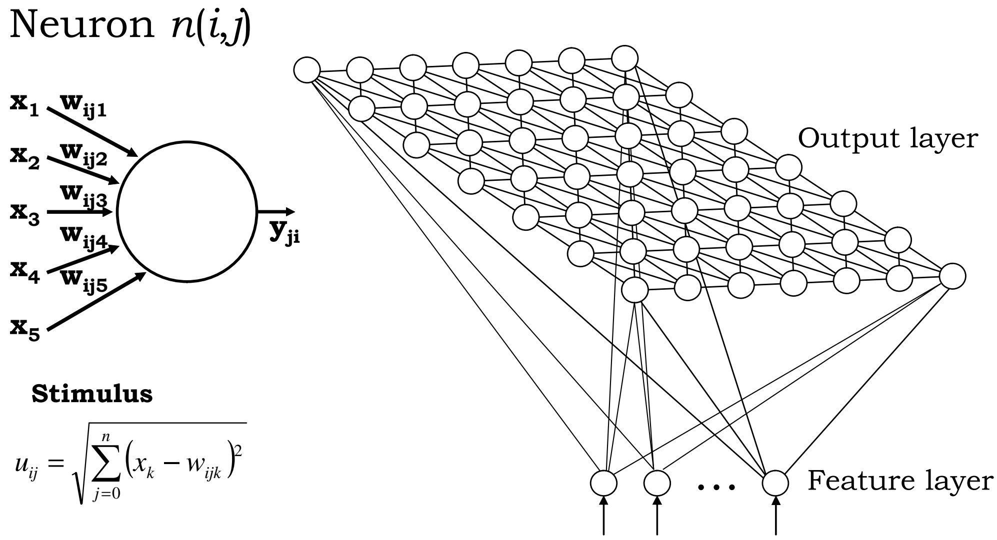
The mapping capabilities of the MLP stem from the nonlinearities used within the nodes. The proposed ANN model uses the hyperbolic tangent function for the hidden units and the sigmoidal function for the output node. Since the mapping range of the sigmoidal function is within the interval [0,1], the output of the MLP model is multiplied by a factor 10 to obtain a [0,10] mapping range.

The network is trained using the standard error backpropagation (BP) rule with momentum term (Rumelhart and McClelland, 1986). According to this algorithm, the MLP uses the set of training patterns to learn the desired behaviour via least squares minimisation of the output error. The algorithm is run for a fixed number of iterations which is manually set to optimise the learning accuracy. Learning via backpropagation is akin to stochastic approximation of the input-output relationship. The learning parameters of the BP algorithm are optimised according to experimental trial and error.

Once the architecture is optimised and the ANN is trained, the system is ready to operate. Table 2 summarises the final MLP structure and BP rule settings.

3.2. Unsupervised artificial neural network model

Kohonen's self-organising feature map (SOM) (Kohonen, 1984) was originally created to reproduce the organisation of biological sensory maps of the brain. This ANN model implements a clustering algorithm that is akin to K-means. Fig. 5 illustrates a typical SOM architecture. Due to its simple



Stimulus

$$u_{ij} = \sqrt{\sum_{j=0}^{n} \left(x_k - w_{ijk}\right)^2}$$

Response

$$f\left(u_{ij}\right) = \begin{cases} u_{ij} = \max\left(u_{kl}, k \in [1, Rows], l \in [1, Columns]\right) \Rightarrow 1 \\ else \Rightarrow 0 \end{cases}$$

Fig. 5: Kohonen's self-organising map

architecture, versatility and ease of implementation, the SOM is the most popular kind of unsupervised ANN system. SOMs found application in several financial domains (Deboeck, 1998).

The SOM is composed of two layers of nodes, namely the feature layer and the output layer.

The feature layer collects the ANN input and forwards it to the neurons of the next layer. The output layer is composed of a two-dimensional grid of processing units. Each neuron measures the similarity between the input pattern and a *reference vector* stored in the values of the incoming weights. Similarity is measured as the Euclidean distance between the reference vector and the input vector.

Neurons of the output layer operate in a competitive fashion, that is, only the unit having the best matching reference vector is allowed to respond to the input pattern (winner-take-all rule).

Learning generates a vector quantiser by adjusting the incoming weights of the winner neuron to resemble more closely the input pattern. Other neurons in the neighbourhood have their weights modified of an amount that is increasingly scaled down as their distance from the winner unit widens. The magnitude of the weight correction factor is controlled via a *neighbourhood function*. In biological systems, competitive activation and neighbourhood learning are obtained respectively via inhibitory and excitatory synapses.

Self-Organising Map Settings	
Input nodes	5
Output nodes	15x15 grid
Initialisation range for weights	[-0.3, 0.3]
Learning Parameters	
Learning trials	10
Learning iterations (τ)	10000
Learning coefficient (at iteration t)	$1-t/\tau$
Neighbourhood function	Gaussian
Spread of gaussian neighbourhood	$15*(1-t/\tau)$

Table 3: SOM settings and learning parameters.

Upon iterative presentation of the input patterns, the ANN self-organises to respond with topologically close nodes to physically similar patterns. Reference vectors cover the input distribution by moving toward the centres of the clusters of training data samples. As learning proceeds, the amount of weight adaptation is decreased to allow finer adjustments of the SOM behaviour. Changes are also made more local by increasing the dampening of the weight correction factor with the distance from the winner neuron. At the end of the process, only the weights of the winner node are adjusted. The final setting of the reference vectors tends to approximate the maximal points of the probability density function of the training data (Kohonen, 1984).

SOMs can be used in model approximation by presenting the network with input vectors composed by input-output pairs of training patterns. The ANN adjusts its behaviour to cluster similar sets of condition-response pairs. During the validation phase, only the input pattern is fed and matched with the corresponding elements of the reference vector (i.e. the condition). The remaining weights of the winner neuron (i.e. the response) define the ANN model response.

Because of the topological organisation of the output layer, neighbouring conditions elicit similar responses, thus ensuring a smooth mapping of the desired relationship. Accordingly, previously unseen data are mapped according to the most similar training examples.

The SOM architecture and the learning parameters are set according to experimental trial and error. For the proposed study, a SOM having an input layer of 5 units (4 monthly economic indicators plus the corresponding US bonds yield) and an output layer of 15x15 units is built. The number of mapping nodes is suggested by the low sampling of the training space, since a large number of neurons ensures a smoother coverage of the unsampled input space. Table 3 summarises the main SOM settings and training parameters.

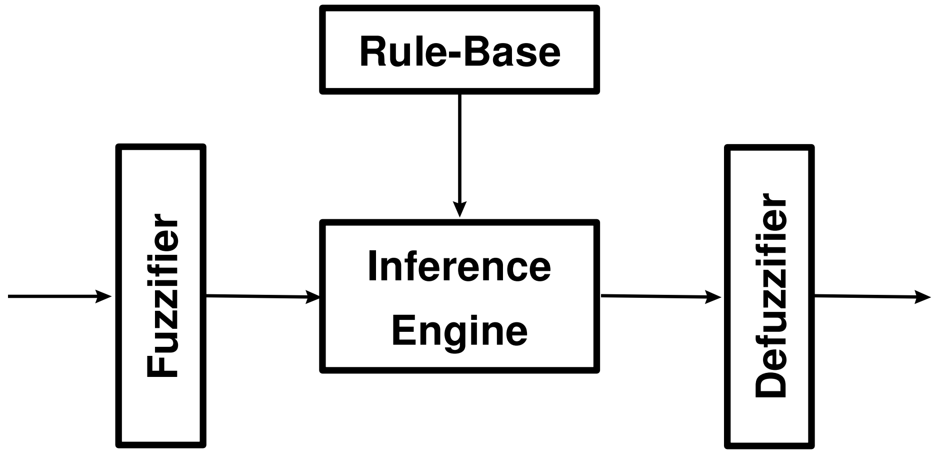


Fig. 6: Mamdani-type fuzzy logic system

3.3. Manually designed fuzzy logic model

A standard Mamdani-type (Mamdani, 1974) FL system is used. The block diagram of this system is shown in fig. 6.

Fuzzy sets are defined via trapezoidal membership functions (MFs), while output defuzzification is performed via the height method (Castellani and Pham, 2002a). Since no expert knowledge is available in the form of fuzzy if-then rules, the FL model is built solely on the basis of the available data samples.

The partition of the input and the output spaces is determined according to experimental trial and error. The space of each of the four input variables is divided into seven evenly spaced linguistic terms. The output space is divided into nine linguistic terms spanning the interval [0, 10].

The rule base (RB) is built by creating a fuzzy rule out of each of the 216 training examples. Rules are generated by associating the fuzzy terms that better match the values of the input variables to the term that better matches the desired output. Duplicate rules are removed, rules having the same input but different output are resolved by choosing the case that best fits the training examples. At the end of the process, the span of each input MF is slightly enlarged to reflect the uncertainty about the space partition.

This procedure generates 110 rules that are then used to forecast the values of the remaining 12-month out of sample 10-year Treasury bonds yields.

3.4. Automatically designed fuzzy logic model

Although the design of the fuzzy model is conceptually straightforward, much effort is required to generate the mapping knowledge base (KB) (i.e., the fuzzy rules and MFs). Keeping the same Mamdani-type FL system used in the previous test, an alternative inductive machine learning approach is investigated for automatic identification of the 10-year Treasury bonds time series.

The generation of FL systems is essentially a search problem, where the solution space is represented by the large number of possible system configurations. Evolutionary algorithms (EAs) (Fogel, 2000) are a class of global search techniques that provide an ideal framework for the task. As well as allowing the optimisation of both the KB and the MFs, EAs only need a small amount of problem domain expertise for implementation.

EAs are modelled on Darwin's theory of natural evolution. This stipulates that a species improves its adaptation to the environment by means of a selection mechanism that favours the reproduction of those individuals of highest fitness. A population of candidate solutions (i.e., FL systems) is iteratively made to evolve until a stopping criterion is met. At the end of the process, the best exemplar is chosen as the solution to the problem.

In EAs, the adaptation of an individual to the environment is defined by its ability to perform the required task. A problem-specific fitness function is used for the quality assessment of a candidate solution. The population is driven toward the optimal point(s) of the search space by means of stochastic search operators inspired by the biological mechanisms of genetic selection, mutation and recombination. Problem-specific operators are often used to speed up the search process.

The EA used in this trial (Pham and Castellani, 2002b) generates Mamdani-type FL systems through simultaneous evolution of the RB and MFs. The algorithm uses the generational replacement reproduction scheme (Fogel, 2000) and an adaptive selection operator (Pham and Castellani, 2002b) that aims at maintaining the selection pressure constant throughout the whole evolution process. A set of crossover and mutation procedures each concerned with a different level of KB optimisation is used, namely, RB optimisation, MFs optimisation, and optimisation of both RB and MFs simultaneously.

Each member of the starting population is initialised with a blank RB and a random partition of the input and output spaces. During the fitness evaluation phase, candidate solutions are tested on the series of training data points. At each step, a solution forecasts the value of the US bonds yield by searching its KB for the rules best matching the set of input conditions. The algorithm creates a new rule if the set of input conditions having the highest matching degree does not lead to an existing rule action. The consequent of the newly generated rule is randomly determined. The aim of the

EA parameters	
Population size	80
Learning trials	10
Learning iterations	500
Crossover rate	1
Mutation rate	0.1
Max number of terms per variable	6
Initialisation parameters	
Number of terms per variable	4
Rule base	empty
Fitness function settings	
Evaluation steps	216
Error measure	root MSE

Table 4: EA parameters.

procedure is to limit the RB growth only to the most relevant instances.

The fitness of the candidate solutions is evaluated as the measure of their root mean square modelling error over the set of training patterns. The lower the error is, the higher the chances are that the solution is selected for reproduction. Each learning cycle, a fitter population is produced through genetic crossover and mutation of the individuals that are selected for reproduction (i.e., the best performing ones). This procedure is repeated until a pre-defined number of iterations has elapsed and the fittest solution of the last generation is picked.

The learning parameters are set according to experimental trial and error. Table 4 summarises the main EA settings.

4. CLASSICAL MODELLING APPROACHES

In this study two classes of traditional models are tested. The first model is an univariate model, in which future values of the variable are predicted only using current and past values of the own variable. For this reason, it belongs to the class of statistical models. The second model uses a set of variables chosen according to economic theories about the nature of the relationship with the variable to be forecast. Since the second model combines economics, mathematics and statistics, it is an example of econometric model. For a brief overview of the two classical models presented, the reader is referred respectively to appendices C and D.

4.1. ARIMA model

The first step to build the ARIMA model is the identification of the data-generating process. There are some statistical rules that help to find out the appropriate specification. In this regards, visual

Variable	ADF Test Statistic	
	Level	First Diff.
US 10y TB yields	-1.300460	-7.138813
PMI index	-3.312909	-6.633664
CPI-yoy	-0.849709	-5.350012
Libor-3m	-1.410705	-5.008327
VIX index	-2.906936	-9.735054
1% Critical Value -3.4612 5% Critical Value -2.5737 10% Critical Value -2.5737		

Table 5: Augmented Dickey-Fuller Unit Root Test.

Dependent variable: Δ(U.S. 10-year Treasury bond yield) Method: Least Squares Number of observations: 213 after adjusting endpoints		
Variable	Coefficient	t-Statistic
constant	-0.016974	2.328155
AR(1)	-0.090818	3.542232
AR(2)	-0.944908	3.186955
MA(1)	0.092203	8.690526
MA(2)	0.99182	-2.505165
S.E. of regression	0.287393	
Durbin-Watson statistic	1.823712	
F-statistic	4.150752	
Prob(F-statistic)	0.002953	
Inverted AR Roots	-0.05+0.97i	-0.05-0.97i
Inverted MA Roots	-0.05+0.99i	-0.05-0.99i

Table 6: Output from ARIMA(2,1,2) model.

inspection of correlograms of the autocorrelation function and of the partial autocorrelation function is often recommended. The order of differentiation is related to the need to work with stationary time series. In many economic variables, first-difference is enough to achieve that objective.

Since the Dickey-Fuller test (Dickey and Fuller, 1979; 1981) indicates that the U.S. 10-year Treasury bonds yield is an integrated variable of first order, the ARIMA model is estimated in first-difference. Following extensive experimental estimations, it is concluded that the ARIMA(2,1,2) is the best model for the available sample in terms of forecast performance and also because of its parsimony of parameters. The augmented Dickey-Fuller unit root test for all the variables is presented in table 5.

The output from the ARIMA estimation is shown in table 6. *AR*(*p*) is the component containing just the *p* lagged dependent variable terms statistically meaningful in the past history of the process. *MA*(*q*) is the disturbance component of the model. All *AR* and *MA* terms have high levels of

Dependent variable: U.S. 10-year Treasury bond yield Method: Leasr Squares Number of observations: 214 after adjusting endpoints		
Variable	Coefficient	t-Statistic
constant	1.511884	2.328155
x_1: Purchasing Managers' Index	0.031313	3.542232
x_2: Core CPI, y-o-y rate of previous period	0.361839	3.186955
x_3: 3-month LIBOR on US dollar	0.454422	8.690526
x_4 Volatility Index of the CBOE	-0.009144	-2.505165
Error of previous period	0.907958	29.40349
R-squared	0.972524	
Adjusted R-squared	0.971863	
S.E. of regression	0.248929	
Durbin-Watson statistic	1.883824	
F-statistic	1472.428	
Prob(F-statistic)	0.000000	

Table 7: Output from the econometric model.

statistical significance. Moreover, the inverted roots of the polynomials have absolute value no greater than one. Before using the estimated equation to forecast the 12-month values ahead of the variable, the performance of an augmented Dickey-Fuller test is used to confirm that the residuals of the equation are white noise disturbances.

4.2. Econometric model

The output from the econometric estimation is shown in table 7. Upon the output of the regression, it is concluded that that all coefficients are statistically significant within the usual standard levels of confidence. The residuals from the regression are a white noise series. R-squared is the coefficient of determination. When multiplied by 100 it represents the percentage of variability in the dependent variable that is explained by the estimated regression equation. For this reason, it is a measure of the strength of the regression relationship.

All the coefficients have the expected signals predicted by economic theory and summarised earlier in this paper. The coefficients of the variables related to economic growth, inflation and reference interest rates are positive indicating a direct relationship with the yields on long-term Treasury bonds. The negative coefficient of the volatility index suggests a negative correlation between the bonds market and the stock exchange condition as it is very often observed.

5. EXPERIMENTAL RESULTS

This section compares the accuracy results obtained using econometric, statistical and AI models for forecasting te US 10-year Treasury bonds yields. In all the cases, the models are fitted using the 216 data samples covering the 18-years span between 1986:1 and 2003:12. The evaluation of the

model	Accuracy	Std. deviation
Fuzzy (manual)	0.3523	-
Fuzzy (learned)	0.3325	0.0872
SOM	0.2693	0.0248
One-step lag	0.2574	-
MLP	0.2480	0.0016
ARIMA(2,1,2)	0.2464	-
Econometric	0.2376	-

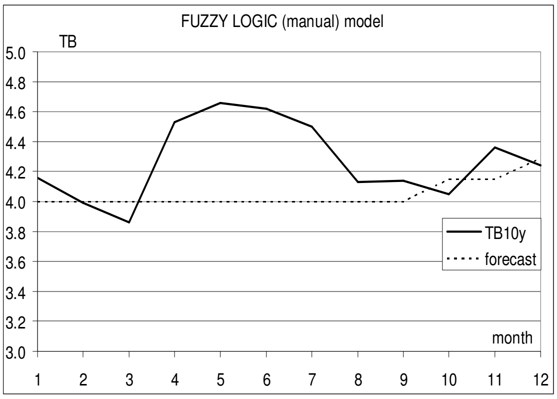
Table 7: RMSE modelling error.

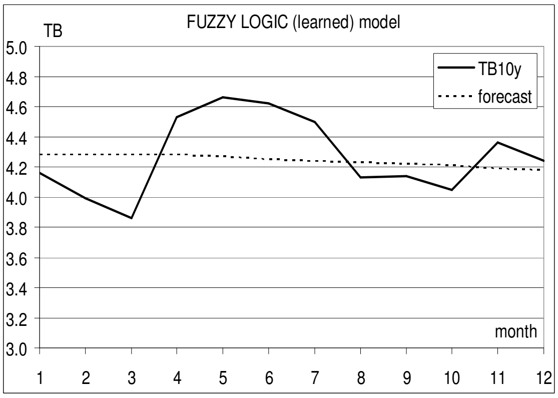
models is based on the RMSE of the 12-month out of sample forecasts (2004:1 to 2004:12). For the machine learned AI models, accuracy results are estimated on the average of 10 independent learning trials. Table 8 gives the accuracy results of the six modelling approaches. For the sake of comparison, table 8 includes also the RMSE of a one-step lag predictor, that is, a basic algorithm that predicts the yields of the US 10-year Treasury bonds from the figure of the previous month.

For each approach, figs. 7a-f show the evolution of the actual Treasury bonds yields and the corresponding forecasts of the models over the 12-month evaluation period. For the machine learning approaches, figs. 7a-f show a sample result.

The two FL modelling approaches give the worst accuracy results. Considering the high standard deviation of the RMSE of the EA-generated models, the difference in accuracy between the two approaches is statistically not significant. However, the automatic method creates more compact solutions. These solutions are characterised by a RB that is on average half the size of the RB of the manually designed FL system.

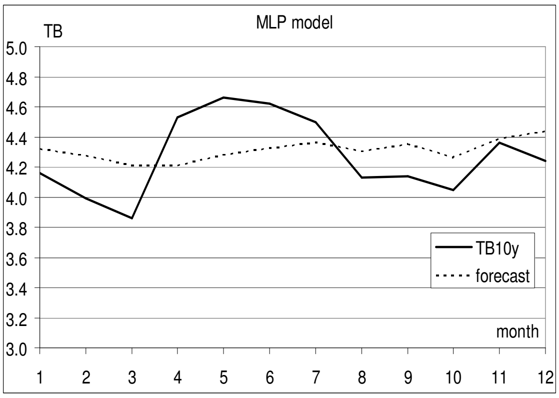
Figs. 7a-b show the behaviour of two sample FL models. The forecast of the manually fitted FL system for the US 10-year Treasury bonds yields is constant throughout most of the year (9 months). During the period that the response is flat, the output of the model is decided exclusively by one rule. Since the overall system behaviour is built by assembling condition-response associations taken from the history of the past years, the mainly flat output of the manually built FL system suggests that insufficient data may have prevented a finer modelling of the desired relationship. Indeed, inspection of figs. 2a-e shows the combination of values that the input variables take during the 12-month evaluation period has little history in the past. Namely, the range of values taken by the CPI and the Libor indexes mostly reflects the history of the last years, while the range of values taken by the PMI and VIX indexes finds correspondence in more remote times.





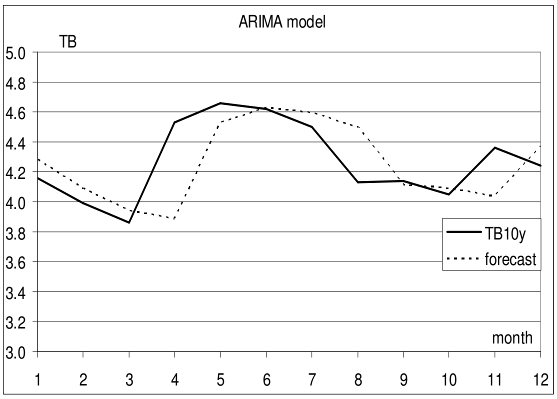
a) manually designed FL model

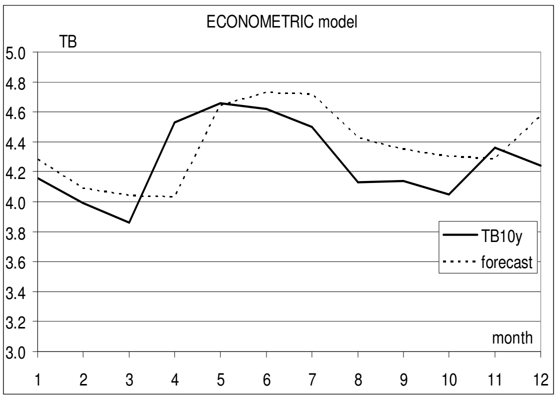
b) EA-generated FL model (sample)



c) MLP model (sample)

d) SOM model (sample)





e) ARIMA model

f) ECM model

Fig. 7: Modelling results

Since the mapping of the EA-generated FL system is also learned from the same data samples, the results of the automatic FL modelling procedure do not improve the results of the manual FL modelling procedure. Similarly to the case of the manually fitted FL model, the output of the sample EA-generated model appears to be dominated by very few rules, roughly modelling the main trend of the US 10-year Treasury bonds yields over the validation span.

The SOM system outperforms the two fuzzy models in terms of learning accuracy and robustness. The superior performance of the SOM model stems from the generalisation capability of the ANN. Although the learning algorithm is equally based on pointwise mapping of input-output training pairs, the action of the neighbourhood function (see subsection 3.2) partially fills the gaps between the centres of the training data clusters with the response of neurons that are not clearly committed to any cluster. The large number of neurons utilised in this experiment is likely to have helped the SOM system to provide a better response to previously unseen input data. Fig. 7d shows the response of a sample SOM model. Analysis of fig. 7d shows that the ANN output resembles a slowly varying interpolation of the desired curve. Also in this case, the lack of historical data allows only the overall trend to be modelled.

The MLP is the AI system that gives the best modelling results. The average accuracy achieved by the MLP solutions improve of about 10% the results obtained by the SOM model. The very small standard deviation indicates the high consistency of the learning procedure. This result is due to the distributed way that this type of ANN uses to store the mapping knowledge. FL and SOM systems cluster similar data patterns into single functional units, respectively rules and neurons, and use some in-built mechanism to generalise to unseen cases, respectively the MF width and the vector quantiser properties of the competitive layer. In MLP systems, the memorisation of each single training pattern affects the setting of all the ANN weights. As a consequence, the overall behaviour is set to best fit the distribution of the whole training data. This "global" fitting of the training data improves the generalisation capability of MLP systems, particularly when a limited training set is available and the "local" data fitting procedure of SOM and FL systems disregards unsampled input conditions.

Fig. 7c shows the response of a sample MLP system throughout the evaluation year. The curve follows more closely the evolution of the US 10-year Treasury bonds yields, even though the quality of the mapping is still quite coarse.

The accuracy of the statistical model is within one standard deviation from the average accuracy of the MLP model. Given that the MLP learning algorithm performs a stochastic approximation of the desired input-output relationship, the equivalence of the modelling results reflects the similar statistical nature of the two modelling approaches. The similarity between the MLP and the statistical model extends also to their global approach to curve fitting. In the case of the ARIMA model, the model is fit by adjusting the global system response through the ARIMA parameters. Fig. 7e shows the output of the ARIMA(2,1,2) model. The curve resembles a one-step lag prediction model.

Finally, the econometric ECM model obtains the best forecasting accuracy. Given the lack of historical data that affected the performance of the AI and pure statistical models, it is likely that the superior performance of the ECM model is due to the embedded problem domain knowledge. Fig. 7f shows the output of the econometric model. Also in this case, the gross system response seems to resemble a one-step lag prediction model.

6. DISCUSSION

The main difficulty of the modelling task results from the sparseness of the data that are used to deduce the models. Indeed, only 216 data points are available to identify the highly complex mapping from the 4-dimensional vector of economic indicators to the US Treasury bonds yields. This lack of historical data puts severly to the test the generalisation capability and the reliability of the modelling algorithms under evaluation. Unfortunately, such situation is not uncommon in the field of financial market prediction, where the completeness of the sample data is restricted within the boundaries of past market fluctuations. Given that some economic indicators are published only on periodical basis (e.g., the monthly PMI), the availability of historical data is further restricted.

The sparseness of the data samples affects the accuracy of the six models. In particular, the two FL system and the unsupervised ANN system give unsatisfactory results in terms of precision of the forecasts and reliability of the learning procedure. The main reason for the failure of these three methods is in the modelling algorithm, which is based on the composition of several local input-output associations that are inferred from the distribution of the training data. Such approach is liable to produce poor prediction results when the input conditions are dissimilar from the cases covered by the training data. It is important to note that, in the case of the two FL systems, the poor prediction results are related to the chosen data-driven induction algorithms. A different modelling approach, such as the encoding of expert knowledge (if available), could produce a FL system capable of entirely acceptable performances.

The MLP and the two classical algorithms share the same global approach to modelling, based on parameteric regression of the functional relationship. Their prediction results clearly improve the results obtained by the FL and the SOM systems. However, due to the lack of data samples, the prediction results are only marginally better than the forecasts made using a simple one-step lag algorithm (see table 8).

The MLP, the ARIMA and the ECM models give similar RMSE results. To assess the statistical significance of the differences between the measures, the prediction accuracy of these three models is compared using the Diebold-Mariano (1995) test.

Given a series and two competing predictions, the Diebold-Mariano test applies a loss criterion

variable		forecasts			squared residuals			Difference of squared residuals		
month	**10Y TB**	**ECM**	**ARIMA**	**MLP**	E_{ECM}^2	E_{ARIMA}^2	E_{MLP}^2	E_{MLP}^2- E_{ARIMA}^2	E_{MLP}^2- E_{ECM}^2	E_{ECM}^2-E_{ARIMA}^2
2004:01	4.16	4.28	4.28	4.32	0.0154	0.0152	0.0263	0.0111	0.0109	0.0002
2004:02	3.99	4.09	4.09	4.28	0.0105	0.0103	0.0830	0.0727	0.0726	0.0002
2004:03	3.86	4.04	3.94	4.21	0.0329	0.0071	0.1216	0.1145	0.0887	0.0258
2004:04	4.53	4.03	3.89	4.21	0.2457	0.4092	0.1030	-0.3062	-0.1427	-0.1635
2004:05	4.66	4.64	4.53	4.28	0.0003	0.0175	0.1418	0.1242	0.1414	-0.0172
2004:06	4.62	4.73	4.63	4.33	0.0118	0.0001	0.0846	0.0846	0.0728	0.0118
2004:07	4.50	4.72	4.60	4.36	0.0485	0.0095	0.0189	0.0094	-0.0295	0.0389
2004:08	4.13	4.43	4.50	4.30	0.0891	0.1357	0.0305	-0.1052	-0.0586	-0.0466
2004:09	4.14	4.35	4.11	4.35	0.0440	0.0007	0.0452	0.0446	0.0013	0.0433
2004:10	4.05	4.30	4.09	4.27	0.0633	0.0016	0.0477	0.0460	-0.0157	0.0617
2004:11	4.36	4.29	4.04	4.39	0.0052	0.1046	0.0008	-0.1038	-0.0044	-0.0994
2004:12	4.24	4.57	4.37	4.44	0.1110	0.0169	0.0390	0.0222	-0.0720	0.0942
							std dva	0.1211	0.0788	0.0712
							mean	0.0012	0.0054	-0.0042
							DM	0.0098	0.0685	-0.0592

Table 9: Comparison of prediction accuracies (Diebold-Mariano test).

(such as squared error or absolute error) and then calculates a number of measures of predictive accuracy that allow the null hypothesis of equal accuracy to be tested. The procedure tests whether the mean difference between the loss criteria for the two predictions is zero using a long-run estimate of the variance of the difference series. The most common formula used to perform the Diebold-Mariano test is the following:

$$DM\left(A,B\right)=\frac{AVERAGE(E_A^2-E_B^2)}{STDVA(E_A^2-E_B^2)} \tag{6.1}$$

where DM is the Diebold-Mariano statistic, A and B are two models, E_A and E_B are their prediction errors, and the average and the standard deviation are calculated over the entire validation span.

Table 9 gives the statistics of the Diebold-Mariano test for the comparisons of the three parameteric regression algorithms. Using a one-tailed test at a level of significance of 0.05, the critical value for rejecting the null hypothesis can be inferred from the Standard Normal Distribution to be equal to 1.645. Since the Diebold-Mariano test for the three parameteric regression algorithms gives results that all are clearly lower than the critical value, the hypothesis that the three models have a similar forecasting accuracy can not be rejected.

As a conclusion, although the three parameteric regression algorithms can not be considered statistically different according to the Diebold-Mariano test, the relative size of the MSE points to a better performance of the econometric model as against the ARIMA and the AI models.

7. CONCLUSIONS AND FURTHER WORK

Six AI and classical algorithms are evaluated on their ability of forecasting the monthly yield of the US 10-year Treasury bonds from a set of four economic indicators. The study compares the MSE of four AI models, namely a manually built and a machine learned FL model, a SOM model and a MLP model, with the MSE of two classical models, namely a statistical ARIMA model an econometric ECM model. 216 monthly data samples from 1986:1 to 2003:12 are used to fit the six models and 12 monthly data samples from 2004:1 to 2004:12 are used to validate the results.

In spite of the long observation period, the 216 data samples cover only sparsely the range of possible market fluctuations, representing thus a challenging test for the reliability and the accuracy of the algorithms under evaluation. Experimental evidence indicates the ECM model has a slight edge over the other algorithms, closely followed by the MLP and the ARIMA model. The better performance of the ECM model is likely due to the problem-specific knowledge that is embedded in the algorithm.

The two FL models failed to provide reliable and accurate forecasts for the US Treasury bonds yields. The main reason for their failure is probably due to the data-driven nature of their modelling algorithms, which in combination with the local mapping of the individual fuzzy rules gave poor results in the presence of conditions far from the training examples. For similar reasons, also the SOM model produced an unsatisfactory performance.

Examination of the prediction results of the six models showed that the AI systems tend to approximate the main trend of the modelling variable. However, the lack of an exhaustive training set of examples prevented the AI systems from capturing more detailed oscillations of the US Treasury bonds yields. Conversely, the two classical systems showed a behaviour more resembling a one-step lagged system.

The MSE obtained by the best models is only marginally better that the MSE produced by a basic one-step lag predictor, that is, by predicting the yields of the US 10-year Treasury bonds from the figure of the previous month. This result underlines the difficulty of the modelling test and, in general, the difficulty of building reliable predictors for financial markets. The conclusions of this study suggest that pure data-driven induction can not fully capture the behaviour of the desired variable. A combination of statistical or machine learning techniques with expert knowledge of the financial markets is likely to provide the best predictive accuracy.

Further work should aim at building more powerful hybrid systems, combining machine learned and statistical information with economic theory and expert knowledge. A viable approach would be to incorporate expert knowledge into a FL framework, either as a complement or a complete replacement of the data-driven model fitting algorithms tested in this study. The main difficulty in this approach is represented by the often problematic process of knowledge elicitation from human experts. An alternative hybrid approach to FL modelling would be to combine the output of different predictors into a final forecast. The main difficulty of this approach concerns the definition of the weighing criterion for the output of the different models.

REFERENCES

Baele, L., Ferrando, A.,Hordal, P.,Krilova, E. Monnet, C. (2004), *Measuring Financial Integration in Euro Area*, ECB Occasional Paper Series, No.14.

Bartlmae, K., Gutjahr, S. and Nakhaeizadeh, G. (1997), *Incorporating prior knowledge about financial markets through neural multitask learning*, Prooceedings of the Fourth International Conference on Neural Networks in the Capital Markets.

Chen, A.S. and Leung, M.T. (2005), *Performance Evaluation of Neural Network Architectures: The Case of Predicting Foreign Exchange Correlations*, Journal of Forecasting, vol. 24, no. 6, pp. 403-420.

Cheng, W., Wagner, L. and Lin, C.H., (1996) *Forecasting the 30-year U.S. Treasury bond with a system of neural networks*, Journal of Computational Intelligence in Finance, vol. 4, pp. 10-16.

Box, G. E. P. and Jenkins G. M. (1976), *Time series analysis: forecasting and control*, Revised edition, Holden-Day, San Francisco, CA.

Clements, M.P. and Hendry, D. F. (1998), *Forecasting Economic Time Series,* Cambridge University Press.

Cox, J. C., Ingersoll, J.E. and Ross, S. A., *A Theory of the Term Structure of Interest rates,* Econometrica, 53(2), 1985.

Daniels, H. Kamp, B. and Verkooijen, W. (1999), *Application of neural networks to house pricing and bond rating*, Neural Computing and Applications, vol. 8, pp. 226–234.

Deboeck, G.J. (1998), *Financial applications of self-organising maps*, Neural Network World, vol. 8, no. 2, pp. 213-241.

Dickey, D., and Fuller, W. (1979), *Distribution of the Estimators for Autoregressive Time Series with a Unit Root*, Journal of the American Statistical Association, vol. 74, pp. 427-431.

Dickey, D., and Fuller, W. (1981), *Likelihood Ratio Tests for Autoregressive Time Series with a Unit Root*, Econometrica, vol. 49, pp. 1057-1072.

Diebold, F. X. and Mariano, R. S. (1995), *Comparing Predictive Accuracy,* Journal of Business and Economic Statistics, vol. 13, pp. 253-263.

Din, A. (2003), *Forecasting interest rates using neural network models*, Geneva Research Collaboration, Technical Report, available at http://www.genevaresearch.org/media/GRC-interest-rate.pdf

Dunis, C.L. and Jalilov, J. (2002), *Neural network regression and alternative forecasting techniques for predicting financial variables*, Neural Network World, vol. 12, Part 2, pp. 113-140.

Engle, R. and Granger, C. (1987), *Co-integration and Error Correction Representation, Estimation and Testing*, Econometrica, vol. 35, pp. 251-276.

Fogel, D. B. (2000), *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*, 2nd ed., IEEE Press, New York.

Goonatilake, S., Campbell, J.A. and Ahmad, N. (1995), *Genetic-Fuzzy systems for financial decision making*, Advances in Fuzzy Logic, Neural Networks and Genetic Algorithms, IEEE/Nagoya-University World Wisepersons Workshop, Furuhashi, T. Ed., Springer, Lecture Notes in Artificial Intelligence, USA, pp. 202-223.

Goonatilake, S. and Khebbal, S. (1995), *Intelligent hybrid systems*, Chichester (England), J. Wiley, New York.

Granger, C.W.J., and Newbold, P. (1986), *Forecasting Time Series*, 2nd ed., Academic Press, New York.

Greene, W. (2003), *Econometric Analysis*, 5th Edition, Prentice Hall.

Hellendoorn, H. and Driankov, D. (1997), *Fuzzy model identification : selected approaches*, Hellendoorn and Driankov eds., Berlin; New York: Springer.

Kohonen, T. (1984), *Self-organisation and associative memory*, Springer, Heidelberg.

Kosko, B. (1993), *Fuzzy thinking*, Hyperion, New York.

Lee, C. C. (1990), *Fuzzy logic in control systems: fuzzy logic controller*, Part I & Part II, IEEE Trans. Syst. Man and Cyb., vol. 20, no. 2, pp. 404-418 & 419-435.

Lippmann, R. P. (1987), *An introduction to computing with neural nets*, IEEE ASSP Mag., pp. 4-22.

Mamdani, E. H. (1974), *Application of fuzzy algorithms for control of simple dynamic plant*, Proc. Inst. El. Eng., vol. 121 nr. 12, pp. 1585-1588.

Mc Culloch, W. S. and Pitts, W. (1943), *A logical calculus of the ideas imminent in nervous activity*, Bulletin of Mathematical Biophisics, vol. 5, pp. 115-133.

Mohammadian, M. and Kingham, M. (1997), *Hierarcical Fuzzy Logic for Financial Modelling and Prediction*, 10th Australian Joint Conference on Artificial Intelligence, Perth, Australia, pp. 147-156.

McNelis, P.D. (2005), *Neural Networks in Finance - gaining predictive Edge in the Market*, Academic Press.

Norgaard, M., Ravn, O., Poulsen, N. K. and Hansen, L. K. (2000), *Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook*, Springer-Verlag UK

Pham, D.T. and Castellani, M. (2002a), *Action Aggregation and Defuzzification in Mamdani-type Fuzzy Systems*, Proc. ImechE, part C, vol. 216, no. 7, pp. 747-759.

Pham, D.T. and Castellani, M. (2002b), *Outline of a New Evolutionary Algorithm for Fuzzy Systems Learning*, Proc. ImechE, part C, vol. 216, no. 5, pp. 557-570.

Pham, D. T. and Li, D. (2005), *Fuzzy systems for modelling, control and diagnosis*, Elsevier Science, Oxford.

Refenes, A. N., Bentz, Y., Bunn, D.W., Burgess, A.N. and Zapranis, A.D. (1997), *Financial time series modelling with discounted least squares backpropagation*, Neurocomputing, vol. 14, pp.123-138.

Rich, E. and Knight, K. (1991), *Artificial Intelligence* 2nd ed, MC Graw - Hill Inc.

Rosenblatt, R. (1959), *Principles of Neurodynamics*, Spartan books, New York.

Rumelhart, D. E. and McClelland, J. L. (1986), *Parallel distributed processing: exploration in the micro-structure of cognition*, vol. 1-2, Cambridge, MIT Press.

Trippi, R. and Turban, E. (1996), *Neural Networks in Finance and Investing*, Irwin Professional (UK), 2nd edition.

Walczak, S. (2001), *An Empirical Analysis of Data Requirements for Financial Forecasting with Neural Networks*, Journal of Management Information Systems, vol. 17, no. 4, pp. 203-222.

White, D. A. and Sofge, A. (1992), *Handbook of intelligent control: neural, fuzzy and adaptive approaches*, Van Nostrand Reinhold, New York.

Zadeh, L. A. (1965), *Fuzzy sets*, Information and Contr., vol. 8, pp. 338-353.

Zadeh, L. A. (1968), *Fuzzy algorithms*, Information and Contr., vol. 12, pp. 94-102.

Zadeh, L. A. (1973), *Outline of a new approach to the analysis of complex systems and decision processes*, IEEE Trans. on Syst. Man and Cybern., vol 3, no. 1, pp. 28-44.

APPENDICES

A. ARTIFICIAL NEURAL NETWORKS

The first studies related to computational models of the brain included the theoretical work carried out in the 40's by (Mc Culloch and Pitts, 1943) and Rosenblatt's *perceptron* (Rosenblatt, 1959), the latter being devised in the mid 50's and inspired by the structure of the retina. Since then, many researchers have focused on modelling the structure and the processing mechanisms of human or animal nervous systems. The common name for all these computational models is ANNs.

ANNs are composed of a certain number of elementary units called *neurons*, organised in layers. Each neuron receives inputs from other neurons or the external world via adjustable connections called *weights*, and maps these inputs to the output space via a *transformation function*. The transformation function can vary widely according to the ANN architecture but is usually common within a layer. The output (*activation*) is then distributed to other units or the external world through other connections.

An ANN is generally divided into three parts. The first part is composed of an *input layer* of neurons. These nodes gather the incoming signals to the network and generally act as a buffer. The second part consists of one or more *hidden layers*. These neurons collect the signals coming from the previous layers and process them via a generally non-linear transformation function. This is usually the stage where the input data are clustered and the partition of the input space is detemined. The *ouput layer* is the last part of an ANN. It collects the signals from the previous layers and processes them to give the final output. The network so far described is a *feed-forward neural network*, where the signal flows just in the forward direction. Further connections are sometimes added to feedback the signal to previous layers. This kind of architecture is called a *recurrent neural network* and is mainly used for prediction and control of dynamic processes. A wide survey of ANN architectures can be found in (Lippmann, 1987).

Thanks to their learning capability, ANNs require no prior knowledge about the task to be performed. Typically, the network undergoes a training phase where the weights of the connections between neurons are adjusted. This procedure modifies the system response by modifying the way the incoming signals to the units are scaled.

In an ANN, association rules are distributed among several neurons and data are processed in parallel layer by layer. Thanks to the non-linear mapping of the individual units, ANNs are capable of modelling any arbitrarily complex function. Moreover, their learning and generalisation capabilities remove the need for time-consuming system identification. However, because of its distributed nature, the expertise is not retrievable after the training has ended. ANNs act similarly to a blackbox of which only the input and the output can be observed.

The accuracy result strongly depends from the setting of the ANN topology and from the quality of the learning process. Designing the optimal ANN structure is a 'black art', as the number of neurons and layers for a certain kind of ANN can vary considerably depending on the application. The chosen configuration is often the result of a time-consuming trial and error process, where several solutions need to be generated and trained before the optimal architecture is found.

B. FUZZY LOGIC SYSTEMS

FL (Zadeh, 1965) can be considered as a broadening of classical symbolic logic of which it keeps the deductive structure.

The main idea is the extension of Aristotle's binary logic with the concept of *degree of truth*. Binary logic constrains a statement to being either true or false, failing often to model the complexity and the uncertainty of a definition. FL partitions the space (*universe of discourse*) into overlapping sets called *linguistic terms* that are expressed through *fuzzy sets* (Zadeh, 1968; Zadeh 1973). A fuzzy set A in a universe of discourse U is defined by the following set of ordered pairs:

$$A = \{(\mathbf{u}, \mu_A(\mathbf{u})) | \mathbf{u} \in U\} \tag{A.1}$$

where $0 \le \mu_A(\mathbf{u}) \le 1$ is the *membership function* (MF) for A.

A *linguistic variable* takes the values of the term set on which it is defined in a certain universe of discourse. The true or false dualism is now substituted by a MF that fills the space between the binary extremes {0,1} with the complete interval [0,1] (Zadeh, 1968; Zadeh 1973). The ordinary set-theoretic operations of classical binary logic can be extended to fuzzy sets (Lee, 1990).

FL finds its natural application in the expression of qualitative knowledge that is naturally imprecise and vague. FL systems are usually composed of four blocks, namely the *fuzzifier*, the *rule base* (RB), the *inference engine* and the *defuzzifier* (Lee, 1990). The RB and the set of input and output MFs are often referred as the *knowledge base* (KB) of the system. The fuzzifier transforms crisp data into fuzzy sets and is the interface between the quantitative sensory inputs and the qualitative fuzzy knowledge.

The core of the fuzzy system is constituted by the rule base and the inference engine. It closely resembles the structure of a standard Expert System (Rich and Knight, 1991) of which it can be considered an extension to the fuzzy domain.

The rule base is composed of fuzzy if-then rules made by an antecedent-consequent pair. The conditions in the antecedent are joined by means of *and*/*or* logical connectives, while the consequent generally expresses one action per rule, since rules involving multiple outputs can always be decomposed into a set of single-output rules (Lee, 1990). The connective *and* is commonly expressed through a fuzzy set intersection operation in the Cartesian product space, while the connective *or* is usually associated with a fuzzy set union operation (Lee, 1990).

A fuzzy rule is implemented by a relation between the universe of discourse of the antecedent and the universe of discourse of the consequent (Lee, 1990).

Example A.1 (Fuzzy rule)

'if **x** is *A* and **y** is *B* then **z** is *C*'

where **x**, **y**, **z** are linguistic variables and *A*, *B*, *C* linguistic terms defined respectively in universes of discourse *U*, *V*, *W*.

A rule base **R** composed of *n* control rules $\mathbf{R}_\lambda$ ($\lambda = 1\ldots$, n) is usually expressed by the union of the *n* fuzzy relations:

$$\mathbf{R} = \cup_\lambda \mathbf{R}_\lambda \qquad \text{(A.2)}$$

When the antecedent of a fuzzy rule is matched with a fuzzified observation, the consequent is activated to a level equal to the truth degree of the antecedent. The rules of inference are usually implemented by extending the classical modus ponens rule to fuzzy sets (*generalised modus ponens*) (Zadeh, 1973; Lee, 1990).

FL keeps the rigorous inferencing structure of classical logic and extends it to deal with imprecise data. When all the variables are defined using binary values, FL coincides with predicate logic.

Each rule maps hyperplanes of the input space onto corresponding regions of the output space. The input-output relationship is therefore expressed through 'patches' in the cartesian product of the input and output space (Kosko, 1993). The extension of these patches depends on the fuzziness of the relationship.

Because of the overlapping boundaries of fuzzy terms, the input data can match the antecedent of more than one fuzzy rule. The system response is therefore the result of the interaction of different individual rule mappings.

The inference engine processes the rules and produces an overall response in the form of a fuzzy set. The defuzzifier must then convert that fuzzy output into a crisp number. Many defuzzification methods are possible, the most commonly used ones are the *Centre Of Gravity* method and the *Mean of Maxima* method (Lee, 1990).

The structure of fuzzy system described so far is the most popular and is often dubbed the *Mamdani model* (Mamdani, 1974). Fuzzy mappings can be used as a qualitative model of the system, thus avoiding the time-consuming and complex stage of analytically modelling unknown dynamics. The fuzzy partition of the input space can fully express the uncertain nature of the expertise, eliminating the need for often complex and arbitrary constraints on the variables. It also makes the system more robust to noise and data corruption since the matching procedure is not bounded by perfect correspondence. Moreover, the graded and overlapping division of the input and output space smoothes the response of the system.

FL modelling is akin to non-parametric basis function regression, where each rule can be thought of as a basis function. If a set of qualitative 'rules of thumb'are available, the human-like nature of fuzzy logic makes it easier for experts to express such knowledge about the system behaviour. Alternatively, when expertise is not available, fuzzy rules can be obtained from experimental observations via machine learning techniques. Different strategies can be used and combined to create or modify fuzzy mappings, that is, new rules can be added or deleted, the input and the output space partitions (i.e. the membership functions) can be modified, or both the operations can be performed simultaneously.

A wide survey and an analysis on fuzzy identification systems is presented in (Hellendoorn and Driankov, 1997).

C. ARIMA MODEL

One of the most popular univariate time series model is the general ARIMA(p,d,q) model, where *p* is the number of autoregressive terms, *d* is the number of differences (order of integration), and *q* is the number of lagged disturbance terms. Its representation form is

$$\theta(L)(1\text{-}L)^d y_t = c + \phi(L)\varepsilon_t \qquad \text{(C.1)}$$

where y_t is the time series, ε_t is the random error, c is a constant term and L is the backshift operator: $Ly_t = y_{t\text{-}1}$.

$\theta(L)$ is the autoregressive operator that is represented as a polynomial in the back shift operator, that is,

$$\theta(L) = 1\text{-}\ \theta_1 L - \theta_2 L^2 - \ldots - \theta_p L^p \qquad \text{(C.2)}$$

finally, $\phi(L)$ is the moving-average operator, represented as a polynomial in the back shift operator, that is,

$$\phi\ (L) = 1\text{-}\ \phi_1 L - \phi_2 L^2 - \ldots - \phi_p L^p \qquad \text{(C.3)}$$

where t indexes time.

D. ERROR CORRECTION MODEL

The error correction model (ECM) (Engle and Granger, 1987) is the econometric method chosen to provide an alternative forecast for the time series of the bonds yields. This method differs from the standard regression model as it includes an error correction term to account for the cointegration issue. The ECM is generally considered by the specialised literature to possess a high predictive accuracy and appropriate to capture both the long and short-term dynamics of a time series.

Despite the variables are integrated of order 1, our choice to estimate the variables in levels within the ECM framework follows the method suggested by Engle and Granger (1987) that preserves the long-run relationship between the variables and takes into account the short-run dynamics implied by the deviations of the variables from their long-term trend.

The following equation is estimated

$$y_t = \beta_0 + \beta_1 x_{1t} + \ldots + \beta_4 x_{4t} + \lambda(y_{t-1} - \hat{y}_{t-1}) + \varepsilon_t \qquad \text{(D.4)}$$

where y_t is the dependent variable, x_{it},…, x_{it} are the independent variables, y_{t-1}-$\hat{y}_{t-1}$ is the modelling error of the previous period, β_0 is a constant term, β_1,…, β_4 are the coefficients of the independent variables, t represents the time step, λ is the speed at which y adjusts to the error in the previous period, and ε_t is the random residual.

An alternative statistical model would have been a VAR (vector autoregressive). In a context where some variables are weakly exogenous, a VAR model has the virtue of obviating a decision as to what variables are exogenous and what variables are not. In our case, some causal effects from the left-hand-side variable to the right-hand-side variables can not be ruled out. However, the forecasting performance of an ad-hoc VAR model estimated using all the five variables over the same period compares poorly with the forecasting performance of the other models.